De-anonymization attack on geolocated datasets

Sébastien Gambs, Marc-Olivier Killijian, Miguel Nuñez del Prado Cortez

De-anonymization attack on geolocated data

Sébastien Gambs
Université de Rennes 1 - INRIA / IRISA
Campus Universitaire de Beaulieu
35042 Rennes, France
Email: sgambs@irisa.fr

Marc-Olivier Killijian[1]
Miguel Núñez del Prado Cortez[1,2]
[1]*LAAS - CNRS*
7 avenue du Colonel Roche, BP 54200, F-31031 Toulouse, France
[2]*Univ. de Toulouse, INSA, LAAS, F-31400 Toulouse, France*
E-mail: {marco.killijian,mnunezde}@laas.fr

***Abstract*—With the advent of GPS-equipped devices, a massive amount of location data is being collected, raising the issue of the privacy risks incurred by the individuals whose movements are recorded. In this work, we focus on a specific inference attack called the de-anonymization attack, by which an adversary tries to infer the identity of a particular individual behind a set of mobility traces. More specifically, we propose an implementation of this attack based on a mobility model called Mobility Markov Chain (MMC). A MMC is built out from the mobility traces observed during the training phase and is used to perform the attack during the testing phase. We design two distance metrics quantifying the closeness between two MMCs and combine these distances to build de-anonymizers that can re-identify users in an anonymized geolocated dataset. Experiments conducted on real datasets demonstrate that the attack is both accurate and resilient to sanitization mechanisms such as downsampling.**

***Keywords*-Privacy, geolocation, inference attack, de-anonymization.**

I. Introduction

With the recent advent of ubiquitous devices and smartphones equipped with positioning capacities such as GPS (Global Positioning System), a massive amount of mobility traces is collected and gathered in the form of geolocated datasets by cellphone companies, providers of location-based services, and developers of smartphone applications. Some of these geolocated datasets are available in public repositories and can be used for research or for industrial purposes (*e.g.*, to optimize the placement of cellular towers, to conduct market and sociological studies or to analyze the flow of traffic inside a city). These datasets are composed of the mobility traces of hundred or thousands of individuals [1], [2], [3], [4], thus raising the issue of the privacy risks incurred by these individuals. For instance, from the movements of an individual it is possible to infer his points of interests (such as his home and place of work) [5], [6], [7], [8], to predict his past, current and future locations [9], [10], or even to infer his social network [11].

In this work, we focus on a particular form of inference attack called the *de-anonymization attack*, by which an adversary tries to infer the identity of a particular individual behind a mobility trace. More precisely, we suppose that the adversary has been able to observe the movements of some individuals during a non-negligible amount of time (*e.g.*, several days or weeks) in the past during the training phase. Later, the adversary accesses a different geolocated dataset containing the mobility traces of some of the individuals observed during the training phase, plus possibly some unknown persons. Then, the objective of the adversary is to de-anonymize this dataset (called the *testing dataset*) by linking it to the corresponding individuals observed during the *training phase*. Note that simply replacing the real names of individuals by pseudonyms before releasing a dataset is usually not sufficient to preserve the anonymity of their identities because the mobility traces themselves contain information that can be uniquely linked back to an individual. In addition, while a dataset can be sanitized before being released by adding spatial and temporal noise, the risk of re-identification through de-anonymization attack nevertheless still exists.

In this paper, we propose a novel method to de-anonymize location data based on a mobility model called Mobility Markov Chain (MMC) [7]. A MMC is a probabilistic automaton, in which each state corresponds to one (or possibly several) Point Of Interest (POIs), characterizing the mobility of an individual and an edge indicates a probabilistic transition between two states (*i.e.*, POIs). Each state can have an semantic label attached to it such as "home", "work", "leisure", "sport", ... A MMC is built out of the mobility traces observed during the training phase and is used to perform the de-anonymization attack during the testing phase. More precisely, the mobility of each individual both from the training and testing sets is represented in the form of a MMC. Afterwards, a distance is computed between possible pairs of MMCs from the training and testing sets in order to identify the closest individuals in terms of mobility. In short, the gist of this method is that the mobility of an individual can act as a signature, thus playing the role of a quasi-identifier [12]. Thus, if the adversary knows Alice and a signature of her mobility (*e.g.*, he has learnt her MMC out of the training set), he can try to identify her by finding a matching signature in the testing set.

The outline of the paper is the following. First, in Section II, we review some related work on de-anonymization attacks and mobility models before briefly introducing in

Section III the background on Mobility Markov Chains necessary to understand our work. Afterwards in Section IV, we present the distance metrics between MMCs that we designed in order to quantify the closeness between two mobility behaviors, while in Section V we describe how to build predictors (which we call *de-anonymizers*) based on these distances to efficiently and accurately de-anonymize location data. Finally, we evaluate experimentally the efficiency of the proposed attack on real geolocated datasets in Section VI before concluding in Section VII.

II. Related work

An *inference attack* corresponds to a process by which an adversary that has access to some data related to individuals (and potentially some auxiliary information) tries to deduce new personal information that was not explicitly present in the original data. For instance, a famous inference attack was conducted by Narayanan and Shmatikov on the "Netflix dataset" [13]. This dataset is a sparse high dimensional data containing ratings on movies of more than 500 000 subscribers from Netflix that was supposed to have been anonymized before its release for the Netflix competition[1]. However, Narayanan and Shmatikov have performed a de-anonymization attack that was able to successfully re-identify more than 80% of the Netflix subscribers by using the Internet Movie Database[2] (IMDB), another database of movie ratings, as auxiliary knowledge.

Inference attacks have also been developed specifically for the geolocated context. For instance, Mulder *et al.* [6] have proposed two methods for profiling users in a GSM network that can also be used to perform a de-anonymization attack. The first method is based on constructing a Markov model of the mobility behavior of an individual while the second considered only the sequence of cell IDs visited. Once POIs have been extracted for each user, an agglomerative hierarchical clustering algorithm is used to group users according to a similarity measure called the *cosine similarity* [14]. Their first method is relatively similar to ours with two major differences due to the fact that their dataset comes from a cellular network. Thus, it relies on the static GSM cells as the states of the Markov model, while we dynamically learn the POIs from the mobility traces of an individual. Therefore, in our setting two individuals do not necessarily have any POI in common, whereas with GSM cells, individuals living in the same area have a high probability of sharing some POIs (*i.e.*, cells). As a consequence, the second main difference between their work and ours is that the transitions are only possible between neighboring GSM cells, as it is impossible to "jump" from one cell to another if they are not adjacent. Their attack was validated through experimentations using cell locations from a MIT Media Lab dataset[3] [15] that includes information such as call logs, Bluetooth devices in proximity, cell tower IDs, application usage and phone status. During the experiments, the authors have observed that if the currently profiled user belongs to a cluster of other similar users, there is a high chance of making a mistake about the identity inferred among all the users of this cluster when performing the de-anonymization attack. The success rate of the re-identification attack varies from 37% to 39% using the Markov model, against 77% to 88% when the sequence of cells visited is used.

Zang and Bolot [8] have performed a study of the top n most frequently visited places by an individual in a GSM network and show how they can act as quasi-identifiers to re-identify anonymous users. Their study was performed on the Call Data Record (CDR) from a nationwide US cellular provider collected over a month and contains approximately 20 millions users. From this dataset, the authors have identified the top n most frequent places for each user at different levels of spatial (*i.e.*, sector, cell, zip code, city, state and country) and time granularity (*i.e.*, day and month). Their inference attack was able to successfully re-identify 35% of the population studied when the adversary has no auxiliary knowledge and even up to 50% when the adversary can use the knowledge of the social network of users as auxiliary information. The social network was constructed by creating a social relationship between two individuals that have called each other at least once in the past. In their analysis, the authors emphasize that the distance between home and work can be an indicator of the privacy level for an individual. In particular, the larger this distance, the higher the risk that this individual can be de-anonymized.

Ma *et al.* [16] have also proposed an inference attack to de-anonymize users in a geolocated dataset along with a metric to quantify the privacy loss of an individual. Two datasets were used in this study, one taken from the Crawdad repository recording the movements of San Francisco YellowCabs [1] and another recording the movements of Shanghai city public buses[4]. Two types of adversary models were considered: the *passive* one, collecting the whereabouts of individuals from a public source (possibly sanitized) and the *active* one that can deliberately participate to the data collection or influence it by his acts in order to gain additional knowledge about the location of some specific individuals. To retrieve the identities of individuals, the authors imagine four different estimators that the adversary can use to measure the similarity between mobility traces (*e.g.*, between the original traces and the sanitized ones). Basically, the attack proposed by Ma *et al.* extracts the signature of the mobility of an individual and evaluate depending on the size of this signature (ranging from 1 to 30 timestamped positions) how it uniquely identifies the

[1] http://www.netflixprize.com
[2] http://www.imdb.com
[3] http://reality.media.mit.edu/dataset.php
[4] To the best of our knowledge, this dataset is not publicly available.

target individual. These methods approach a success rate of de-anonymization of 80% to 90% on the San Francisco YellowCabs dataset and between 60% and 70% on the Shanghai dataset, and this even when the data is sanitized through the addition of spatial noise. However, contrary to our work, these inference attacks were conducted on the whole dataset (there was no split between a training set and a testing set). In particular, the authors assume that both type of adversaries (*i.e.*, passive and active ones) pick the information they need to build the mobility model from the same dataset on which the success of the de-anonymization attack is tested. This induces an overly strong bias in the re-identification results obtained with this approach.

The work in [17] also focuses on re-identifying users of geolocated datasets. These experiments have been conducted on two datasets. The first dataset contains the GPS traces of 24 users from the city of Borlange recorded in a two-year period (1999-2001)[5] while the second one is due to Nokia[6] and is composed of the GPS traces of 150 users from the city of Lausanne recorded over a year. In this work, the pair of POIs "home/work" is used as pseudo-identifiers to de-anonymize users. First, a variant of the k-means algorithm is used to extract POIs from the mobility traces. Then, the POI in which the individual considered stays the most often between 9PM to 9AM is identified as "home" while the POI in which the individual stays the most often between 9AM to 5PM is labelled as "work". The training phase consists in applying this method on the raw data to extract the pairs "home/work" for all individuals, and then to conduct the same attack on sampled traces in order to assess how much the pair "home/work" can still be inferred even when the dataset released has been sanitized by applying downsampling. The success of this method depends on how the number of sampled traces have been generated. Indeed, the authors have proposed several sampling schemes whose bias towards selecting home/work locations or other POIs can be parametrized. The authors have shown that when 100 samples are observed, the de-anonymization rate is approximately 70% for the Nokia dataset and 67% for the Borlange one. Unfortunately, as most of the previous works presented, this study does not split the available data into a training and testing set during the evaluation of the success of the method but rather generates the samples that will constitute the testing set directly from the training one. This introduces a major bias in the evaluation of the techniques, which we further discuss in Section VI-C.

The work of Xiao *et al.* [18] relies on the notion of Semantic Location Histories (*SLH*) to compute the similarity between users. In a nutshell, a SLH is simply the sequence of semantic locations frequently visited by an individual. Like several previous approaches, this work first uses a hierarchical clustering algorithm to extract POIs out of mobility traces. Then, using as external knowledge a database to associate semantics to a location, each POI is associated with a semantic tag for each level of the hierarchy (*e.g.*, "Italian restaurant" and then "restaurant" at an upper level). Finally, the SLH is computed by analyzing the sequence of POIs visited by a user and taking into account their semantic labels. The similarity measure designed by the authors is based on the notion of *maximal travel match*, which counts the number of similar semantic locations visited (not necessarily in the same order) by two different SLHs within a predefined time interval. This metric is computed for each layer of the cluster hierarchy before being summed over all possible layers, possibly by weighting the influence of a particular level (*i.e.*, the deeper the level, the bigger the influence). Finally, the proposed approach was evaluated on the Geolife dataset [19]. Contrary to previous work, the success of the de-anonymization attack is quantified in terms of the *normalized discounted cumulative gain*, a metric originated from information retrieval [20]. In a nutshell, the objective of this metric is to rank all the possible candidates to de-anonymization with respect to how close their mobility is to the behavior of the user considered. In the experiments conducted, the success of the attack as measured by this metric was between 0.7 and 0.9. Basically the closer this value is to 1, the more effective the attack is. Note that, due to the different metric that was used, this method is not directly comparable to other previous works.

Finally, Shokri *et al.* [21] infered the correspondence between pseudonymized traces of 40 randomly chosen users of the YellowCabs dataset [1], in which each position is a cell of 8×5 grid over the San Francisco Bay area, and user profiles represented in the form of hidden Markov model. Their attack computes a matching probability between pseudonymized traces and user profiles by using the classical Forward-Backward algorithm [22]. As the objective of this attack was individual tracking, their results are not directly comparable to our work.

In this section, we have reviewed the previous work on de-anonymization attacks in the geolocated context. Thereafter, we will present our novel approach to de-anonymization. More precisely, we first introduce how to model the mobility of an individual in the form of a MMC, before describing how to measure the similarity between two MMCs using the distances we propose. Finally, we demonstrate experimentally how to use these distance metrics to perform a de-anonymization attack.

III. Mobility Markov Chain

A *Mobility Markov Chain* (MMC) [7] models the mobility behavior of an individual as a discrete stochastic process in which the probability of moving to a state (*i.e*, POI) depends only on the previously visited state and the probability

[5]http://icapeople.epfl.ch/freudiger/borlange.zip

[6]To the best of your knowledge, this dataset is not publicly available.

distribution on the transitions between states. More precisely, a MMC is composed of:

- A *set of states* $P = \{p_1, \dots p_n\}$, in which each state is a frequent POI (ranked by decreasing order of importance), with the exception of the last state p_n that corresponds to the set composed of the union of all infrequent POIs. POIs are learned by running a clustering algorithm on the mobility traces of an individual. These states are associated to a location, and generally they also have an intrinsic semantic meaning. Therefore, semantic labels such as "home" or "work" can often be inferred and attached to them.
- *Transitions*, such as $t_{i,j}$, represent the probability of moving from state p_i to state p_j. A transition from one state to itself is possible if the individual has a non-null probability from moving from one state to an occasional location before coming back to this state. For instance, an individual can leave home to go to the pharmacy and then come back to his home. In this example, it is likely that the pharmacy will not be extracted as a POI by the clustering algorithm, unless the individual visits this place on a regular basis.

Note that several mobility models based on Markov chains have been proposed in the past [7], [23], including the use of hidden Markov models for performing inference attacks [24]. In a nutshell, building a MMC is a two steps process. During the first phase, a clustering algorithm is run to extract the POIs from the mobility traces. For instance in the work of Gambs *et al.* [7], a clustering algorithm called Density-Joinable Cluster (*DJ-Cluster*) was used (we rely on the same algorithm in this work), but of course other clustering algorithms are possible. In the second phase, the transitions between those POIs are computed.

DJ-Cluster takes as input a trail of mobility traces and three parameters: the minimal number of points $MinPts$ needed to create a cluster, the maximum radius r of the circle within which the points of a cluster should be contained and a distance d at which neighboring clusters are merged into a single one. DJ Cluster works in three phases. During the first phase, which corresponds to a preprocessing step, all the mobility traces in which the individual is moving (*i.e.*, whose speed is above a small predefined value) as well as subsequent static redundant traces are removed. As a result, only static traces are kept. The second phase consists in the clustering itself: all remaining traces are processed in order to extract clusters that have at least $MinPts$ points within a radius r of the centre of the cluster. Finally, the last phase merges all clusters that have at least a trace in common or whose medoids are within d distance of each other. Once the POIs (*i.e.*, the states of the Markov chain) are discovered, the probabilities of the transitions between states can be computed. To realize this, the trail of mobility traces is examined in chronological order and each mobility trace is tagged with a label that is either the number identifying a particular state of the MMC or the value "unknown". Finally, when all the mobility traces have been labeled, the transitions between states are counted and normalized by the total number of transitions in order to obtain the probabilities of each transition. A MMC can either be represented as a transition matrix or as a graph in which nodes correspond to states and arrows represent the transitions between along with their associated probability. When the MMC is represented as a transition matrix of size $n \times n$, the rows and columns correspond to states of the MMC while the value of each cell is the probability of the associated transition between the corresponding states.

IV. Distances between Mobility Markov Chains

In this section, we propose two different distances quantifying the similarity between two Mobility Markov Chains. These distances are based on different characteristics of the MMCs and thus give different but complementary results. We will rely on these distances in the following sections to perform the de-anonymization attack.

A. Stationary distance

The intuition behind the *stationary distance* is that the distance between two MMCs corresponds to the sum of the distances between the closest states of both MMCs. In order to compute the stationary distance, the states of the MMCs are paired in order to minimize this distance. As a result, it is possible that a state from the first MMC is paired with several states of the second MMC (this is especially true if the MMCs are of different size). Furthermore, the computation of the stationary distance heavily relies on the stationary vectors of the MMCs. In a nutshell, the stationary vector of a MMC is a column vector V, obtained by multiplying repeatedly a vector initialized uniformly V_{init} by the MMC transition matrix M until convergence (*i.e.*, until the distribution of values in this vector reaches the stationary distribution of the MMC).

The stationary distance is directly computed from the stationary vectors of two MMCs (hence its name). More precisely, given two MMCs, M_1 and M_2, the stationary vectors, respectively V_1 and V_2, of each model are computed. Afterwards, Algorithm 1 is run on these two stationary vectors. For each state in V_1, the algorithm searches for the closest state in V_2 (lines 5 to 11) and then multiplies the distance between these two states by the corresponding probability of the stationary vector of the state of V_1 currently considered (line 12).

Once the algorithm has taken into account all states from V_1, the current value computed represents the distance from M_1 to M_2 ($distance_{AB}$ in line 1, Algorithm 2). This distance is not symmetric as such and therefore in order to symmetrize it, Algorithm 1 is called once again, but on V_2 and V_1 in order to obtain the distance from M_2

to M_1 ($distance_{BA}$ of line 2, Algorithm 2). The result is made symmetrical by computing the average of these two distances (line 3, Algorithm 2).

Algorithm 1 Stationary_distance(V_1, V_2)

1: $distance = 0$
2: **for** $i = 1$ to n_1 (the number of nodes in V_1) **do**
3: $MinDistance = 100000$ kilometers
4: Let p_i be the i^{th} node of V_1
5: **for** $j = 1$ to n_2 (the number of nodes in V_2) **do**
6: Let p_j be the j^{th} node of V_2
7: $CurrentDistance =$ Euclidean_Distance(p_i, p_j)
8: **if** $(CurrentDistance < MinDistance)$ **then**
9: $MinDistance = CurrentDistance$
10: **end if**
11: **end for**
12: $distance = distance + Prob_{V_1}(p_i) \times MinDistance$
13: **end for**
14: **return** $distance$

Algorithm 2 Symmetric_stationary_distance(V_1, V_2)

1: $distance_{AB} =$ Stationary_distance(V_1, V_2)
2: $distance_{BA} =$ Stationary_distance(V_2, V_1)
3: $distance = (distance_{AB} + distance_{AB})/2$
4: **return** $distance$

B. Proximity distance

The intuition behind the *proximity distance* is that two MMCs should be considered as close if they share "important" states. For instance, if two individuals share both their home and place of work they should be considered as being highly similar. Moreover, the importance of a state is directly proportional to the frequency at which it is visited. Therefore, the first states ordered by decreasing order of importance are compared, then the second ones, then the third ones, and so on. The proximity score obtained for sharing the first states is considered twice as important as the score for sharing the second states, which is itself twice as important as the sharing of the third states, and so forth.

Given two MMC models M_1 and M_2 (ordered in a decreasing manner with respect to their stationary probabilities), this distance is parametrized by a threshold Δ and a $rank$. The objective of the rank is to quantify the importance of matching two states at a specific level. In particular, the higher is the value of the $rank$, the bigger is the weight that will be given to these POIs. For instance for the first pair of POIs, we have set $rank = 10$.

Algorithm 3 starts by verifying for each pair of nodes between M_1 and M_2 if the Euclidean distance between them is less than the threshold Δ (line 8). If this condition is met, the value of $rank$ is added to the score value (line 9). Afterwards, $rank$ is divided by two (lines 11-14). Once all the pair of nodes have been processed, the global distance is set to be the inverse of the global score if this score is non-null (lines 17-19). Otherwise, the distance outputted is set to a large value (*e.g.*, 100 000 kilometers).

Algorithm 3 Proximity_distance(V_1, V_2)

1: Sort the states of V_1 by decreasing order of frequency
2: Sort the states of V_2 by decreasing order of frequency
3: $score = 0$, $rank = 10$
4: **for** $i = 1$ to $\min(n_1, n_2)$ **do**
5: Let p_a be the i^{th} node of V_1
6: Let p_b be the i^{th} node of V_2
7: $distance =$ Euclidean_distance(p_a, p_b)
8: **if** $(distance < \Delta)$ **then**
9: $score = score + rank$
10: **end if**
11: $rank = rank/2$
12: **if** $(rank = 0)$ **then**
13: $rank = 1$
14: **end if**
15: **end for**
16: $distance = 100000$
17: **if** $(score > 0)$ **then**
18: $distance = 1/score$
19: **end if**
20: **return** $distance$

The stationary distance is composed of the sum of the Euclidian distances of some pairing of the states while the proximity distance is completely different as it is based on the semantics behind the MMCs. Indeed, the first state in the model is inferred as being very representative of the mobility of an individual (*e.g.*, home), the second as quite representative (*e.g.*, the place of work), and two individuals are considered as very similar if they share these two places. Thereafter, we will see how these distance can be used to build de-anonymizers and how their diversity can be leveraged and combined to enhance the success of the de-anonymization attack.

V. De-anonymizers

In this section, we rely on the two distances proposed in the previous section to build statistical predictors in order to perform a de-anonymization attack. We call such a predictor, a *de-anonymizer* in reference to its main objective. A de-anonymizer takes as input the MMC representing the mobility of an individual and tries to identifies within a set of anonymous MMCs, the one that is the most similar (*i.e*, the closest in terms of distance). For example, a de-anonymizer may learn from the training set a MMC representing the mobility of Alice and later look for the presence of Alice in the testing set. A de-anonymizer can be based on one distance or a combination of them.

The *minimal distance* de-anonymizer considers that in each row, the MMC with the minimal distance (*i.e.*, the column) is the individual corresponding to the identity of the row. We have considered two instanciations of this de-anonymizer, one with the stationary distance and the other with the proximity distance. The *stat-prox* de-anonymizer behaves exactly like the minimal stationary distance de-anonymizer, except when the stationary distance is above a given threshold and the proximity distance is below its maximum value (*i.e.*, 100 000 kilometers). The intuition is that if the minimal stationary distance is very small, we should use it. Otherwise, we rely on the minimal proximity distance unless it gives no conclusive result, in which case we roll back to the minimal stationary distance.

An overview of the process of *de-anonymization attack* over geolocated datasets used in our experiments is illustrated in Figure 1. Considering a particular geolocated dataset, we first sort the mobility traces of each user in a chronological order. Then, for each user his trail of mobility traces is split into two disjoint trail of traces of same size, one for the training set and one for the testing set as explained in Section VI-A. The former is part of the auxiliary knowledge gathered by the adversary while the latter is the ground truth we use to assess the success of the attack. For each of this dataset, we learn a MMC for each trail of mobility traces. With respect to the MMCs learnt from the training set, the adversary knows the correspondence between these models and the corresponding identities of the users. Afterwards, the distances described in Section IV are used to compute a distance matrix between the MMC models resulting from the training and testing sets. Subsequently, using this distance matrix as input to one of the de-anonymizer described in this section, the objective of the de-anonymization attack is to map back the users of the testing set to their true identities by linking their models in the testing set to the corresponding ones in the training set. Finally, the success rate of the attack is computed by measured the ratio between the number of correct predictions over the total number of guesses.

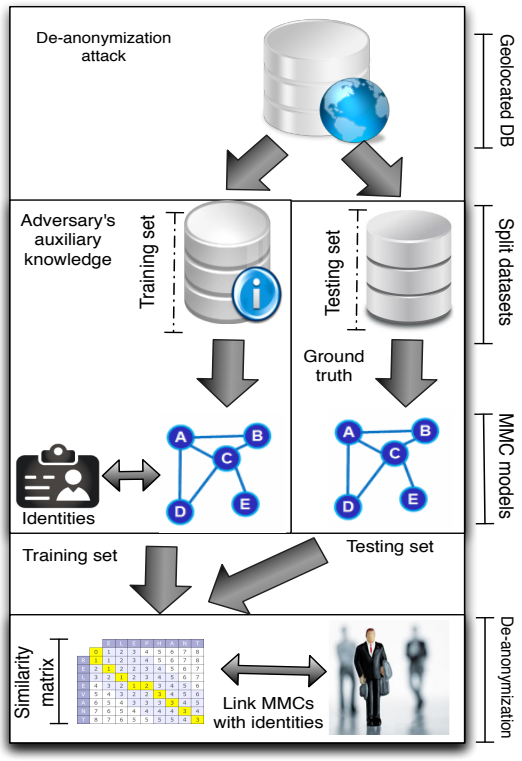

Figure 1. Overview the de-anonymization attack process.

VI. Experiments

We evaluate the efficiency of the de-anonymization attack pictured in the previous section on six different datasets described in Section VI-A. Then, we report the results of those experiments that were conducted by using the distances and de-anonymizers described in the previous sections. More precisely, we evaluate the accuracy of the de-anonymization attack relying on either a single predictor or a combination of them (Section VI-B). Finally, in Section VI-C, we compare our work with the performance reported in related works using biased experiments.

A. Description of datasets

The datasets used in the experiments are the following:

1) The *Geolife dataset* [19] has been gathered by researchers from Microsoft Asia and consists of GPS traces collected from April 2007 to October 2011, mostly in the area of Shanghai city. This dataset contains the mobility traces of 178 users captured at a very high rate of 1 to 5 seconds.
2) The *Nokia dataset* [2] is the result of a data collection campaign performed the city of Lausanne for 200 users started in September 2009 and that lasted for more than two years. The rate at which the location has been sampled varies depending on the current battery level.
3) The *Arum dataset* [4] is composed of the GPS traces of 5 researchers sampled at a rate of 1 to 5 minutes in the city of Toulouse from October 2009 to January 2011.
4) The *San Francisco Cabs dataset* [1] contains GPS traces of approximately 500 taxi drivers collected over 30 days, between May and July 2008, in the San Francisco area.
5) The *Borlange dataset* [17] has been collected as a part of traffic congestion experiment over two years from 1999 to 2001. The public version of this dataset contains the GPS traces of 24 vehicles.

In the following, we first focus on the Geolife dataset in order to analyze and understand the behaviour of the

de-anonymizers and distances. More precisely, for each individual of this dataset, we split his trail of mobility traces into two disjoint parts of approximately the same size. The first half of the original data forms the training set, and will be used as the adversary background knowledge, while the second half constitutes the testing set on which the de-anonymization attack is conducted. For instance, if the original trail of one individual is composed of n mobility traces $\{mt_1, mt_2, \ldots, mt_n\}$, it will be split into a training set $\{mt_1, mt_2, \ldots, mt_{\frac{n}{2}}\}$ and a testing set $\{mt_{\frac{n}{2}+1}, mt_{\frac{n}{2}+2}, \ldots, mt_n\}$ (for illustration purpose we assume that n is an even number). Therefore, the objective of the adversary is to de-anonymize the individuals of the testing set by linking them to their corresponding counterparts in the training set.

B. Measuring the efficiency of de-anonymizers

To measure the success rate of the proposed de-anonymizers, we have sampled the Geolife dataset at different rates and observed the influence of the sampling on the success rates of the de-anonymizers. Figure 2 shows that the success rate of the attack with the minimal stationary distance and the minimal proximity distance varies from 20% to 40%, but that the best performing predictor is stat-prox with results ranging from 35% to 40%.

At this point of the experiments, it seems important to be able to compare precisely the de-anonymizers. Indeed, the success rate of a de-anonymization attack is not the only aspect that should be considered. For instance, for an adversary a possible strategy is to focus on weak individuals that offer a high probability of success for the attack rather than being able to de-anonymize the entire dataset. Measuring the probability of success of the inference attack for a given individual is similar to have some kind of confidence measure for a given de-anonymization candidate. Deriving this confidence measure is quite intuitive for our de-anonymizers. Indeed, for the minimal distance ones, the smaller is the distance, the higher the confidence.

In order to compare the performance of the de-anonymizers, we rely on the notion of *Receiver Operating Characteristic* (ROC) curve [25]. In a nutshell, a ROC curve is a graphical plot representing the sensitivity (*i.e.*, as measured by the true positives rate versus false positives rate) for a classifier. The intuition behind this metric is that between two de-anonymizers achieving the same success rate, one should favor the one displaying the highest confidence. Henceforth, the following so-called ROC curve (Figure 3) shows the true positives rate (TPR) versus the false positives rate (FPR) for the best performing de-anonymizers, with the candidates sorted by ascending distance. This ROC curve further confirms that the stat-prox de-anonymizer is the best alternative among the de-anonymizers we propose.

Our approach performs fairly well for the Geolife dataset as the achieved success rate is between 35% and 45% for the stat-prox de-anonymizer. In order to further validate the approach, we applied it on the Nokia dataset. This dataset has 195 users, among which we can generate a "valid" MMC composed of more than one POI for 157 users using the parameters described previously. As shown on Figure 4, the success rate varies between 35% and 42%, with the best score obtained again by the stat-prox de-anonymizer.

C. Fair comparison with prior work

In this section, we have presented various experiments on de-anonymization attacks that lead to the definition of an heterogeneous de-anonymizer called stat-prox, which obtains a success rate between 42% and 45% on different datasets. While at first glance, this performance may seem to be poorer than the one achieved by the predictors of Ma *et al.* [16], which goes up to 60% to 90%, we believe that these results are not directly comparable because we clearly differentiate between the training set and the testing set, while these authors perform the learning and the testing on the same dataset, thus inducing a strong experimental bias.

Indeed, our mobility models are built out of the training set, which is disjoint from the test set, whereas one of the adversary model of Ma *et al.* directly extracts mobility traces forming the test set from the training set. Moreover, in our case, the training data is temporally separated from the test data (*i.e.*, the training and the test have been recorded at different non-overlapping periods of time) because the whole dataset has been split into two temporally disjoint parts, whereas the second adversary model of Ma *et al.* picks the information it uses to de-anonymize within the same period as the test data is recorded. Therefore, our approach is quite different from them as our attack consists first in collecting mobility data from an individual, before later *in the future* trying to identify this individual in a so-called anonymized dataset, while their attack aims at gathering location data at the same time at which the de-anonymization attack occurs. In addition, one important parameter of their attack is the number of timestamped location data collected, which can be compared to the number of states we have in our mobility model. On average and depending on the dataset considered, we have between 4 and 8 states per MMC, which correspond to a compact representation of the mobility behavior of an individual. When restricted to such limited of information in terms of the number of timestamped location data, the attacks proposed by Ma *et al.* do not perform well, with a de-anonymization rate between 10% to 40%.

For comparison purpose, we conducted the de-anonymization attack without separating the training and testing sets. The results obtained by related work and for the stat-prox de-anonymizer in this setting using different datasets are summarized in Figure 5. These experiments are, as expected, so biased that they lead to a success rate close to 100% for all the datasets. Once again, *we are not pretending* that our de-anonymization attack

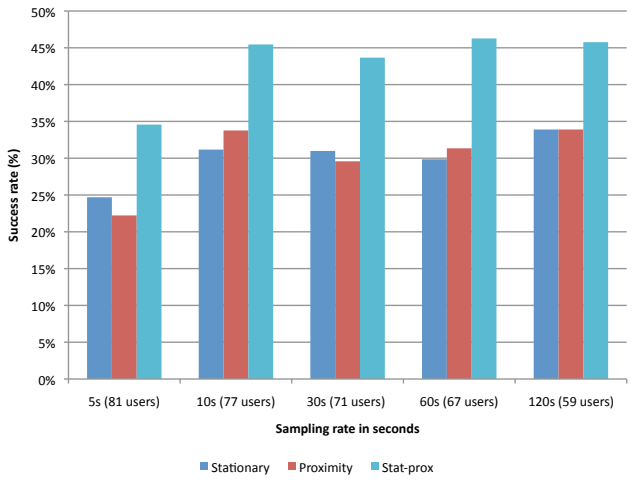

Figure 2. Success rate of the de-anonymizers on the Geolife dataset.

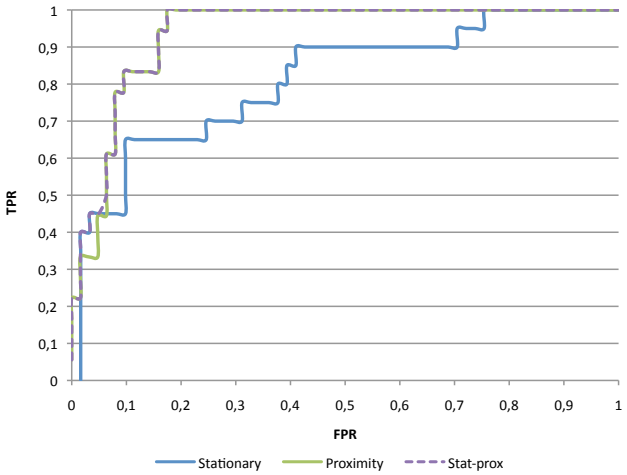

Figure 3. ROC curve for the Geolife dataset.

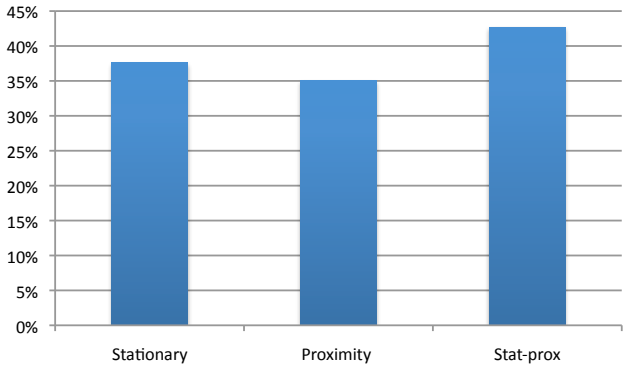

Figure 4. Success rate of the de-anonymizers on the Nokia dataset.

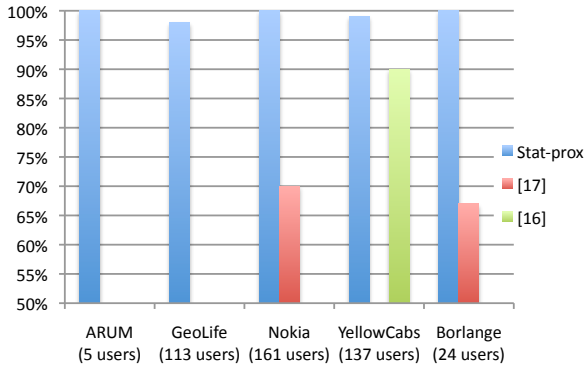

Figure 5. Success rate of the stat-prox de-anonymizer when the training and testing sets are the same.

would achieve a success rate of nearly 100% and beat all the previous methods if tested in the same conditions (for instance in some settings the test set was only a subset of the training set as it was sampled from it). We are merely pointing out that for fairness issues, it is important to compare de-anonymizers using the same setting and that in order to reduce the experimental bias, the training and testing set should be clearly separated (which is not the case in almost all the previous works).

VII. Conclusion

In this paper, we have demonstrated that geolocated datasets gathering the movements of individuals are particularly vulnerable to a form of inference attack called the de-anonymization attack. More precisely, we have shown that the de-anonymization attack can re-identify with a high success rate the individuals whose movements are contained in an anonymous dataset provided that the adversary can used as background information some mobility traces of the same individuals that he has been able to observe during the training phase. Out of these traces, the adversary can build a MMC that models in a compact and precise way the mobility behavior of an individual. We designed novel distances quantifying the similarity between two MMCs and we described how these metrics can be combined to build de-anonymizers. The de-anonymization attack is very accurate with a success rate of up to 45% on large-scale real datasets and this even if the mobility traces are sanitized by downsampling them (*e.g.*, every 2 minutes instead of every 10 seconds). We are planning to extend the current work by following several avenues of research. For instance, one of our research objective will be to discover among different clustering algorithms, the one that best fits our needs while being also robust and stable with respect to small changes in the inputs (*e.g.*, small spatial and temporal perturbation). In a different direction, we will also explore

how more complex geo-sanitization mechanisms, such as spatial cloaking techniques or mix zones, can help to reduce the success rate of the attack.

ACKNOWLEDGMENT

This work was partially supported by LAAS, CNRS, and the "security and privacy for location-based services" activity of EIT ICT labs.

REFERENCES

[1] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "CRAWDAD data set epfl/mobility (v. 2009-02-24)," Downloaded from http://crawdad.cs.dartmouth.edu/epfl/mobility, February 2009.

[2] N. Kiukkonen, "Data collection campaign," Nokia Research Center, Lausanne, Switzerland, Tech. Rep., December 2009.

[3] Y. Zheng, X. Xie, and W.-Y. Ma, "Geolife: A collaborative social networking service among user, location and trajectory," in *IEEE Data Engineering Bulletin*, vol. 33, no. 2, Beijing, P.R. China, March 2010, pp. 32–40.

[4] M. Killijian, M. Roy, and G. Trédan., "Beyond San Francisco cabs: building a *-lity mining dataset," in *Procs. of Workshop on the Analysis of Mobile Phone Networks (NetMob)*, Cambridge, MA, USA, May 2010, pp. 75–78.

[5] J. Krumm, "Inference attacks on location tracks," in *Pervasive Computing*, vol. 4480, Toronto, Canada, June 2007, pp. 127–143.

[6] Y. De Mulder, G. Danezis, L. Batina, and B. Preneel, "Identification via location-profiling in GSM networks," in *Procs. of the 7th ACM Workshop on Privacy in the Electronic Society (WPES '08)*, Alexandria, VA, USA, October 2008, pp. 23–32.

[7] S. Gambs, M.-O. Killijian, and M. Núñez del Prado Cortez, "Show me how you move and I will tell you who you are," in *Transactions on Data Privacy*, vol. 2, no. 4, August 2011, pp. 103–126.

[8] H. Zang and J. C. Bolot, "Anonymization of location data does not work: A large-scale measurement study," in *Procs. of ACM Mobicom*, Las Vegas, NV, USA, September 2011, pp. 145–156.

[9] M. C. Gonzalez, C. A. Hidalgo, and Albert-Laszlo, "Understanding individual human mobility patterns," in *Nature*, vol. 453, New Orleans, LA, USA, June 2008, pp. 779–782.

[10] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," in *Science*, vol. 327, no. 5968, New Orleans, LA, USA, February 2010, pp. 1018–1021.

[11] L. Jedrzejczyk, B. Price, A. Bandara, and B. Nuseibeh, "I know what you did last summer: risks of location data leakage in mobile and social computing," Department of Computing Faculty of Mathematics, Computing and Technology The Open University, Milton Keynes, UK, Tech. Rep., November 2008.

[12] C. Bettini, X. S. Wang, and S. J. jodia, "Protecting privacy against location-based personal identification," *Privacy and Security Support for Distributed Applications*, vol. 3674, pp. 185–199, Novembre 2005.

[13] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Procs. of the 2008 IEEE Symposium on Security and Privacy*, Washington, DC, USA, May 2008, pp. 111–125.

[14] P. Tan, M. Steinbach, and V. Kumar, *Introduction to data mining*, ser. Pearson International Edition. Pearson Addison Wesley, Boston, 2006.

[15] N. Eagle and A. Sandy Pentland, "Reality mining: sensing complex social systems," in *Personal and Ubiquitous Computing*, vol. 10, London, UK, March 2006, pp. 255–268.

[16] C. Y. Ma, D. K. Yau, N. K. Yip, and N. S. Rao, "Privacy vulnerability of published anonymous mobility traces," in *Procs. of the 16th annual international conference on Mobile computing and networking*, New York, NY, USA, 2010, pp. 185–196.

[17] J. Freudiger, R. Shokri, and J.-P. Hubaux, "Evaluating the privacy risk of location-based services," in *Procs. of the 15th international conference on Financial Cryptography and Data Security*. Berlin, Germany: Springer-Verlag, February 2012, pp. 31–46.

[18] X. Xiao, Y. Zheng, Q. Luo, and X. Xie, "Finding similar users using category-based location history," in *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*. New York, NY, USA: ACM, November 2010, pp. 442–445.

[19] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.-Y. Ma., "Understanding mobility based on GPS data." in *In Proceedings of ACM conference on Ubiquitous Computing*, vol. ACM Press, Seoul, Korea, September 2008, pp. 312–321.

[20] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of ir techniques," *ACM Trans. Inf. Syst.*, vol. 20, no. 4, pp. 422–446, Oct. 2002. [Online]. Available: http://doi.acm.org/10.1145/582415.582418

[21] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying Location Privacy," in *Procs. of IEEE Symposium on Security and Privacy (S&P)*, 2011.

[22] S. J. Russell, P. Norvig, J. F. Candy, J. M. Malik, and D. D. Edwards, *Artificial intelligence: a modern approach*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1996.

[23] D. Ashbrook and T. Starner, "Learning significant locations and predicting user movement with GPS," in *Procs. of the 6th IEEE International Symposium on Wearable Computers*, vol. 7, no. 5, Sardina, Italy, February 2003, pp. 275–286.

[24] Z. Yan, D. Chakraborty, C. Parent, S. Spaccapietra, and K. Aberer, "SeMiTri: a framework for semantic annotation of heterogeneous trajectories," in *Procs. of the 14th International Conference on Extending Database Technology*, New York, NY, USA, March 2011, pp. 259–270.

[25] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, June 2006.